

Project #08– Implementing a TreeSort Method

Due: Wednesday, December 11th

You must upload a single ZIP file containing all of your JAVA source code files on the course MOODLE site before the start of the lecture on the due date. Include screencaps of your output from testing your program a few times (the Snipping Tool is handed to capture portions of your screen).

Points will be deducted from late submissions unless previously cleared with me.

Download the Project08Template.ZIP file. This will be the starting point for your project.

The package includes a package containing the implementation of the binary search tree. There is a declared static **treeSort** method in the **BinaryTree** class that you will have to implement.

A tree sort uses the properties of a binary search tree to sort a set of data items. The algorithm is as follows:

1. FOR each element of array
 - 1.1 Insert the element into a binary search tree
2. Recursively Traverse the binary search tree IN ORDER, adding the element of each visited node into the array in next available spot

Think about how you want to implement the method. The 1st step is a straightforward application of using a loop to iterate through the array and inserting the elements into a binary search tree.

The 2nd step will be trickier and there are a few different ways to approach this.

- If you use an array, you will have to find a way to communicate the next available index number between each recursive method call.
- You could use another data structure to temporarily hold the data, and once the entire traversal is done, copy the items back into an array

You are free to take either approach. Once done, run the driver program. DO NOT MODIFY the driver program. It should output the sorted data set.

THINK carefully about how you want to approach the problem. Comment your code to explain your logic.